

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-249658  
(43)Date of publication of application : 17.09.1999

(51)Int.Cl. G10H 7/02  
G10L 3/00  
G10L 5/02

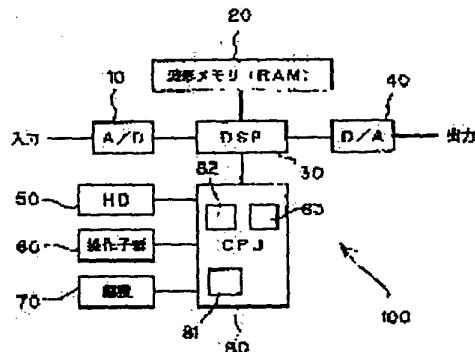
(21)Application number : 10-055387 (71)Applicant : ROLAND CORP  
(22)Date of filing : 06.03.1998 (72)Inventor : SARUHASHI TAKASHI

## (54) WAVEFORM GENERATOR

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To form an output waveform so that its pronounced voice is naturally heard by compressing or extending 1st and 2nd waveform parts in a time base direction by mutually different formats based on stored waveform data.

**SOLUTION:** A waveform memory 20 has a waveform data storing area, a 1st parameter storing area, a 2nd parameter storing area, and a consonant address storing area and a DSP part 30 reads out an original waveform stored in the memory 20 and forms an output waveform having a pitch or the like converted from the pitch or the like of the original waveform. A CPU part 80 calculates a pitch change rate based on sound level information outputted from a keyboard 70 and a reference pitch stored in the 2nd parameter storing area and sends the calculated value to a DSP part 30 together with sound volume information calculated based on velocity information outputted from the keyboard 70 and a set value set up in an operator 60 to form an output waveform corresponding to a depressed key. In this case, the DSP part 30 can set up the patch change rate, a compression/extension ratio and a formant ratio as mutually independent values.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-249658

(43) 公開日 平成11年(1999)9月17日

(51) IntCl.<sup>6</sup>  
G 1 0 H 7/02  
G 1 0 L 3/00  
5/02

識別記号

F I  
G 1 0 H 7/00 5 2 1 Q  
G 1 0 L 3/00 E  
5/02 J

審査請求 未請求 請求項の数6 O L (全 25 頁)

(21) 出願番号

特願平10-55387

(22) 出願日

平成10年(1998)3月6日

(71) 出願人 000116068

ローランド株式会社

大阪府大阪市北区堂島浜1丁目4番16号

(72) 発明者 織橋 隆

大阪市北区堂島浜1丁目4番16号 ローランド株式会社内

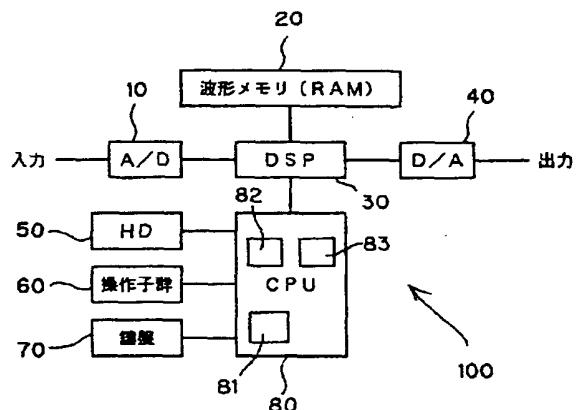
(74) 代理人 弁理士 山田 正紀 (外1名)

(54) 【発明の名称】 波形生成装置

(57) 【要約】

【課題】 雜音のような振る舞いをする部分を含む、言葉等を表す波形に基づいて、その波形とは音高等が異なる出力波形が再生される場合であっても、音が自然に聞こえる波形生成装置を提供する。

【解決手段】 言葉等を表す波形をあらわす波形データを記憶する波形メモリ20と、波形メモリ20に記憶された波形データに基づいて、子音部と母音部とで異なる様様で時間軸方向に圧縮もしくは伸長された波形信号を生成するDSP部30とを備えている。



**【特許請求の範囲】**

**【請求項1】** 第1の波形部分と第2の波形部分が時間的に住み分けながら混在することが許容された波形をあらわす波形データを記憶する記憶手段と、前記記憶手段に記憶された波形データに基づいて、前記第1の波形部分と前記第2の波形部分とで相互に異なる態様で時間軸方向に圧縮もしくは伸張された波形信号を生成する波形生成手段とを備えたことを特徴とする波形生成装置。

**【請求項2】** 前記記憶手段が、相互に異なる第1の性質および第2の性質のうちの第1の性質を持つ第1の波形部分と第2の性質を持つ第2の波形部分が時間的に住み分けながら混在することが許容された波形をあらわす波形データを記憶するものであることを特徴とする請求項1記載の波形生成装置。

**【請求項3】** 前記記憶手段が、前記波形データとともに、該記憶手段に記憶された波形データが表す波形上の、第1の波形部分と第2の波形部分とを識別する識別データを記憶するものであることを特徴とする請求項1または2記載の波形生成装置。

**【請求項4】** 前記波形生成手段が、前記第1の波形部分と前記第2の波形部分とで、圧縮もしくは伸張を行わないことを1つの態様に含む相互に異なる態様で時間軸方向に圧縮もしくは伸張された波形信号を生成するものであることを特徴とする請求項1記載の波形生成装置。

**【請求項5】** 前記記憶手段が、声の波形をあらわす波形データを記憶するものであって、前記第1の波形部分および前記第2の波形部分が、それぞれ、子音部分および母音部分であることを特徴とする請求項1記載の波形生成装置。

**【請求項6】** 前記記憶手段が、時系列的に並ぶ複数の楽音の波形をあらわす波形データを記憶するものであって、前記第1の波形部分および前記第2の波形部分が、それぞれ、個々の楽音の非周期的部分および周期的部分であることを特徴とする請求項1記載の波形生成装置。

**【発明の詳細な説明】****【0001】**

【発明の属する技術分野】本発明は、電子楽器などに用いられる、波形データを記憶し、記憶した波形データに基づいて、その波形データが有する音高等が変換された音高等を有する波形を表す波形信号を生成する波形生成装置に関する。

**【0002】**

【従来の技術】従来より、マイク等を用いて入力された波形データを原波形として記憶し、鍵盤が押下されること等によって音高が指定され、指定された音高に相当する変換率で原波形のピッチを変換し出力波形を再生して発音する電子楽器が知られている。このような電子楽器に例えばフルートの音を原波形として記憶させると、様々な鍵を押下することによって、フルートの音色を有す

る様々な音高の出力波形を得ることができるので、フルートの音色を有する鍵盤楽器として演奏を楽しむことができる。

**【0003】**

【発明が解決しようとする課題】しかし、言葉や打楽器の音等のように立ち上がり部に雑音のような非周期的な振る舞いをする部分を含む原波形に基づいて、前述のような電子楽器に用いられている波形生成装置を用いて出力波形を生成すると、発音されたときの音が不自然に聞こえる出力波形になるという問題がある。

**【0004】** 本発明は、上記の事情に鑑み、雑音のような非周期的な振る舞いをする部分を含む、言葉等を表す波形に基づいて、その波形とは音高等が異なる出力波形が再生される場合であっても、発音されたときの音が自然に聞こえる出力波形が形成される波形生成装置を提供することを目的とする。

**【0005】**

【課題を解決するための手段】上記目的を達成する本発明の波形生成装置は、第1の波形部分と第2の波形部分が時間的に住み分けながら混在することが許容された波形をあらわす波形データを記憶する記憶手段と、記憶手段に記憶された波形データに基づいて、第1の波形部分と第2の波形部分とで相互に異なる態様で時間軸方向に圧縮もしくは伸張された波形信号を生成する波形生成手段とを備えたことを特徴とする。

**【0006】** 本発明の波形生成装置は、上記記憶手段が、相互に異なる第1の性質および第2の性質のうちの第1の性質を持つ第1の波形部分と第2の性質を持つ第2の波形部分が時間的に住み分けながら混在することが許容された波形をあらわす波形データを記憶するものであっても良く、上記記憶手段が、上記波形データとともに、記憶手段に記憶された波形データが表す波形上の、第1の波形部分と第2の波形部分とを識別する識別データを記憶するものであっても良い。

**【0007】** また、本発明の波形生成装置は、上記波形生成手段が、上記第1の波形部分と上記第2の波形部分とで、圧縮もしくは伸張を行わないことを1つの態様に含む相互に異なる態様で時間軸方向に圧縮もしくは伸張された波形信号を生成するものであっても良い。ここでいう「態様」には、時間軸方向への圧縮もしくは伸張の方式や、時間軸方向への圧縮もしくは伸張の率も含まれている。

**【0008】** 更に、本発明の波形生成装置は、上記記憶手段が、声の波形をあらわす波形データを記憶するものであって、上記第1の波形部分および上記第2の波形部分が、それぞれ、子音部分および母音部分であっても良く、上記記憶手段が、時系列的に並ぶ複数の楽音の波形をあらわす波形データを記憶するものであって、上記第1の波形部分および上記第2の波形部分が、それぞれ、個々の楽音の非周期的部分および周期的部分であっても

良い。

【0009】

【発明の実施の形態】以下、本発明の実施形態について説明する。図1は本発明の波形生成装置の一実施形態を示すブロック図である。この波形生成装置100は、1単語あるいは複数単語からなるフレーズを形成する言葉や、楽器で一連のフレーズの演奏を行った演奏楽音等であるフレーズ波形を原波形として記憶し、その原波形を所定のピッチ変化率で変換したり、原波形の時間軸の圧縮伸長を行ったりした出力波形を生成する装置である。このピッチ変化率は、後述するように、鍵盤の鍵が押下されることによって選択され、その選択されたピッチ変化率で変換されたピッチを有する出力波形が再生される。鍵盤の様々な鍵が押下されることにより、その様々な鍵に対応する様々なピッチ変化率それぞれによって1つの原波形が変換され、様々な出力波形が形成される。これにより、鍵盤楽器として演奏を楽しむことができる。また、この波形生成装置では、後述するようにフォルマントや、出力波形全体を発音するのに要する時間がピッチとは独立に設定され、その設定値に基づいて原波形のフォルマント等が変換されて出力波形が再生される。

【0010】この波形生成装置100には、マイクなどを介して入力されるアナログ信号をデジタル信号に変換するA/D変換器10と、A/D変換器10で得られたデジタル信号を表す波形データ等を記憶する波形メモリ(RAM)20が備えられており、波形メモリ20は以下に説明する、波形データ記憶領域、第1パラメータ記憶領域、第2パラメータ記憶領域および子音アドレス記憶領域を有する。

【0011】図2は、波形メモリ20の波形データ記憶領域を表す模式図である。この波形データ記憶領域は、16進数表示で8000番地（以下では番地はすべて16進数表示であるものとする）毎にセグメントで区切られており、各セグメントには、1単語あるいは複数単語からなるフレーズを形成する言葉や、楽器で一連のフレーズの演奏を行った演奏楽音であるフレーズ波形の波形データが記憶される。波形wave0を表す波形データwave data0は、波形データ記憶領域内の相対的な番地で0000番地から記憶され、波形wave1を表す波形データwave data1は0800番地から記憶され、波形wave2を表す波形データwave data2は1000番地から記憶される。以下同様である。

【0012】これらの波形のうちの1つの波形が原波形として選択され、その波形に基づいて出力波形が形成される。上述したように、原波形はフレーズ波形であり、原波形が言葉を表す波形である場合には、原波形のうちの、子音に相当する部分は非周期的な振る舞いをし、母音に相当する部分は周期的な振る舞いをする。また、原

波形が楽音を表す波形である場合には、演奏の開始時の音や打楽器の音等のような非周期的な振る舞いをする部分を含む場合がある。以下では、原波形が表す音の種類に関わらず、言葉を表す波形に準えて、原波形のうちの非周期的な振る舞いをする部分のことを子音部と称し、原波形のうちの周期的な振る舞いをする部分のことを母音部と称する。

【0013】波形データ記憶領域に記憶されている波形データは、後述するような、ユーザによる編集処理に用いられ、出力波形を形成するために必要なパラメータ等が編集・作成される。編集処理では、まず、ソフトウェアによって波形データの先頭から順にゼロクロス位置が求められる。次に、ソフトウェアによって求められたゼロクロス位置を参考に、ユーザによって、子音部および母音部の範囲が指定され、波形データを、波形データが1ピッチ分毎に区切られた区分に区切る位置が指定される。これらの指定に基づいて、区分の先頭のアドレスや区分毎のピッチ等が自動的に計算される。また、波形データが有する複数のピッチを代表する基準ピッチがユーザによって波形データ毎に指定される。

【0014】なお、子音部のピッチは理論的には任意の値でよいので、子音部を区分に区切る位置は任意であるが、ピッチがあまり大きな値である場合には、音がブルブルと不自然に聞こえる現象が生じることが経験上知られているので、子音部を区分に区切る位置の間隔に限界値が設定されていても良い。また、計算処理等の便宜という観点からは、子音部を区分に区切る位置は、子音部の各区分の各ピッチが、子音部の直後に存在する区分のピッチと同じピッチになるように指定されるのが望ましく、以下ではそのような指定がされているものとする。

【0015】上述した編集処理によって得られたパラメータ等が、第1パラメータ記憶領域、第2パラメータ記憶領域および子音アドレス記憶領域に記憶される。図3は、波形メモリ20の第1パラメータ記憶領域を表す模式図である。この第1パラメータ記憶領域は、800番地毎にセグメントで区切られており、波形wave0に対応するパラメータは第1パラメータ記憶領域内の相対的な番地で0000番地から記憶され、波形wave1に対応するパラメータは第1パラメータ記憶領域内の相対的な番地で0800番地から記憶され、波形wave2に対応するパラメータは1000番地から記憶される。以下同様である。

【0016】第1パラメータ記憶領域には、各セグメントの先頭に、波形データの最終のアドレスendadrが記憶され、その後、波形データが1ピッチ分毎に区切られた区分それぞれの先頭のアドレスである区分先頭アドレスsaddrと、各区分のピッチspitchとが交互に記憶される。第1パラメータ記憶領域内の相対的な番地で0000番地には波形データwave data0の最終のアドレスendadrが記憶される。0

001番地には波形データwave data 0の最初の区分の区分先頭アドレスsadr s 0が記憶され、0002番地には波形データwave data 0の最初の区分のピッチpitch 0が記憶され、0003番地には波形データwave data 0の2番目の区分の区分先頭アドレスsadr s 1が記憶される。以後、奇数番地には区分先頭アドレスが記憶され、偶数番地には1番地前に記憶されている区分先頭アドレスに対応する区分のピッチが記憶される。

【0017】なお、以下では説明の便宜上、区分pitch 0という表記を用いて、波形データの、アドレスsadr s 0からアドレスsadr s 1の直前までの部分を表す場合がある。他の区分についても同様である。図4は、波形メモリ20の第2パラメータ記憶領域を表す模式図である。この第2パラメータ記憶領域には、各波形毎の基準ピッチが記憶され、この基準ピッチと、鍵盤の鍵毎に対応付けられているピッチとの比に基づいて、ピッチ変化率が決定される。

【0018】第2パラメータ記憶領域内の相対的な番地の0000番地には波形wave 0に対応する基準ピッチorg. pitch 0が記憶され、0001番地には波形wave 1に対応する基準ピッチorg. pitch 1が記憶され、0002番地には波形wave 2に対応する基準ピッチorg. pitch 2が記憶され、0003番地には波形wave 3に対応する基準ピッチorg. pitch 3が記憶される。以下同様である。

【0019】図5は、波形メモリ20の子音アドレス記憶領域を表す模式図である。この子音アドレス記憶領域には、波形データ記憶領域に記憶されている波形データに子音部が含まれている場合に、その子音部の、先頭のアドレスである子音開始アドレスと末尾のアドレスである子音終了アドレスとが記憶される。図5には、各波形データに複数の子音部が含まれている場合が示されている。これらの子音開始アドレスおよび子音終了アドレスが、本発明にいう識別データに相当する。

【0020】子音アドレス記憶領域は80番地毎にセグメントで区切られており、波形wave 0に含まれる各子音部の子音開始アドレスと子音終了アドレスは、子音アドレス記憶領域内の相対的な番地で00000番地から交互に記憶され、波形wave 1に含まれる各子音部の子音開始アドレスと子音終了アドレスは、子音アドレス記憶領域内の相対的な番地で00080番地から記憶され、波形wave 2に含まれる各子音部の子音開始アドレスと子音終了アドレスは、子音アドレス記憶領域内の相対的な番地で00100番地から記憶される。以下同様である。

【0021】子音アドレス記憶領域内の相対的な番地で00000番地には1番目の子音部の子音開始アドレスv start 1が記憶され、00001番地には1番目の子音部の子音終了アドレスv end 1が記憶され、0

0002番地には2番目の子音部の子音開始アドレスv start 2が記憶され、00003番地には2番目の子音部の子音終了アドレスv end 2が記憶される。以下、偶数番地には子音開始アドレスが記憶され、奇数番地には子音終了アドレスが記憶される。

【0022】図1に戻って説明を続ける。波形生成装置100には、波形メモリ20に記憶されている原波形を読み込んで、原波形が有するピッチ等が変換されたピッチ等を有する出力波形を形成するDSP部30と、デジタル信号をアナログ信号に変換するD/A変換器40が備えられている。なお、DSP部30には、後述する各種のレジスタ等として使用されるワーキングメモリと、ピッチ等の変換方法等を記述したプログラムを記憶するプログラムメモリが含まれる。また、この波形生成装置100には、多くの波形データ等を記憶する大容量の記憶装置であるハードディスク(HD)50と、以下に述べるような各種のスイッチ等を含む操作子群60と、MIDIキーボードである鍵盤70が備えられている。鍵盤70は押下された鍵に対応するノート情報(音高を表す音高情報note\_no.、音量を表すベロシティ情報velocity、および押鍵・離鍵を表すオンオフ情報on/off)を出力する。

【0023】波形生成装置100には、DSP部30の制御や、操作子群60の設定状態の検出・処理や、鍵盤70からのノート情報の検出・処理等を行うCPU部80が備えられており、CPU部80には、後述する各種のレジスタ等として使用されるワーキングメモリ81と、CPU部80用のプログラムを記憶しておくメモリ82と、DSP部30用のプログラムを記憶しておくメモリ83が含まれる。CPU部80に記憶されたDSP部30用のプログラムは、電源オン時にCPU部80からDSP部30のプログラムメモリに転送される。

【0024】鍵盤70の鍵が押下されると、CPU部80により、鍵盤70から出力された音高情報note\_no.と、図4に示す第2パラメータ記憶領域に記憶されている基準ピッチに基づいて、ピッチ変化率が計算され、鍵盤70から出力されたベロシティ情報velocityと、後述する、操作子群60の設定値とに基づいて計算された音量情報とともにDSP部30へ送られる。DSP部30は、CPU部80から送られてきたピッチ変化率および音量情報に従って、鍵盤70の押下された鍵に対応する出力波形を形成する。また、DSP部30は、2つの鍵それぞれに対応する2つの出力波形を同時に形成することもできる。

【0025】一般に、与えられたパラメータに応じて、あらかじめ用意されている原波形を変換して1つの出力波形を形成する装置はボイスモジュールと呼ばれる。従って、DSP部30は、入力されたピッチ変化率や音量情報に応じて原波形を変換して出力波形を形成するという意味においてボイスモジュールに相当する。また、2

つの出力波形を同時に形成するという意味において、D S P部30は2つのボイスモジュールに相当する。以下では説明の便宜上、D S P部30に2つのボイスモジュールが備えられているものとして説明する場合がある。また、以下では1つの鍵に対応する1つの出力波形が再生されて得られる一連の音のことをボイスと称する。

【0026】本実施形態におけるD S P部30では、原波形に基づいて出力波形を形成する際に、ピッチ変化率と、以下説明する圧縮伸長率と、以下説明するフォルマント比率とを互いに独立に設定することができる。ここで圧縮伸長率とは、原波形が発音されるのに要する時間と、出力波形が発音されるのに要する時間との比率のことである。例えば、原波形が言葉を表す波形である場合には、D S P部30によって形成された出力波形が表す言葉は、圧縮伸長率が一定である限り、出力波形が表す声の高さに関わらず一定の速さで話されているように聞こる。

【0027】またフォルマント比率とは、原波形のフォルマントと、出力波形のフォルマントとの比率のことであり、このフォルマント比率は、原波形が表す楽器等の共鳴体としてのサイズと、出力波形が表す楽器等の共鳴体としてのサイズとの比率を表している。例えば、原波形が楽器の楽音を表す波形である場合には、D S P部30によって形成された出力波形が表す楽音は、フォルマント比率が一定である限り、出力波形が表す楽音の音高に関わらず一定の共鳴体としてのサイズを有する楽器によって演奏された楽音が再現される。圧縮伸長率およびフォルマント比率は操作子群60によって設定され、その設定値がC P U部80からD S P部30へ送られる。

【0028】上述のように、C P U部80によって、ピッチ変化率、音量情報、圧縮伸長率およびフォルマント比率がD S P部30へ送られるが、その際に、C P U部80によって、ピッチ変化率、音量情報、圧縮伸長率およびフォルマント比率に周期的な変化が自動的に付加されてD S P部30へ送られる。これによってビブラート演奏やうなりのような発音状態が作られる。以下では、この自動的に付加される周期的な変化のことを「自動周期変化」と称する。この自動周期変化の程度は操作子群60によって設定される。

【0029】D S P部30は以下のよう4つの発音モード(MONO1モード、MONO2モード、POLY1モード、POLY2モード)を有している。以下の説明で「レガート演奏」とは、鍵盤70の鍵を1つ押下したままで、他の鍵を更に押下する演奏方法のことである。MONO1モードは、すべての演奏時点において常に高々1つのボイスが発音されるモードであり、レガート演奏が行われた場合には、新たに押下された鍵に対応するボイスが、それまでに発音されていたボイスに代わって発音され、その新たなボイスが、原波形の先頭位置に対応する音から発音されるモードである。

【0030】MONO2モードは、MONO1モード同様、すべての演奏時点において常に高々1つのボイスが発音されるモードであり、レガート演奏が行われた場合には、新たに押下された鍵に対応するボイスが、それまでに発音されていたボイスに代わって発音され、その新たなボイスが、鍵が押下された時点で発音中の音に対応する原波形上の位置と同じ位置に対応する音から発音されるモードである。

【0031】POLY1モードは、レガート演奏が行われた場合に、押下されている鍵のうち最初に押下された鍵および2番目に押下された鍵それぞれに対応する2つのボイスが発音されるモードであり、MONO1モード同様に、2番目に押下された鍵に対応するボイスが、原波形の先頭位置に対応する音から発音されるモードである。

【0032】POLY2モードは、POLY1モード同様、レガート演奏が行われた場合に、最初に押下された鍵および2番目に押下された鍵それぞれに対応する2つのボイスが発音されるモードであり、MONO2モード同様に、2番目に押下された鍵に対応するボイスが、最初に押下された鍵に対応するボイスが発音中の音に対応する原波形上の位置と同じ位置に対応する音から発音されるモードである。

【0033】なお、POLY1モードおよびPOLY2モードにおいて、2つの鍵が押下されたままで更に3つ目以上の鍵が押下された場合には、その3つ目以上の鍵に対応するノート情報は無視される。また、上述した、POLY2モードにおいてレガート演奏が行われた場合の発音態様のことを以下では、一方のボイスから他方のボイスへのリンク発音と称する。

【0034】図1に示す波形生成装置100は録音処理(REC)、編集処理(EDIT)および再生処理(PALAY)それぞれを行う3つの動作モードを有しており、録音処理を行う動作モードでは、A/D変換手段10を介して入力された波形データが、図2に示す波形データ記憶領域に記憶される。この録音処理を行うモードは、本発明の主題には直接は関係しない。このため、この録音処理を行うモードに関するこれ以上の説明は省略する。編集処理を行う動作モードでは、上述した、図2に示す波形データ記憶領域に記憶された波形データに基づく、パラメータ等の編集作成が行われ、図3～図5に示す、第1パラメータ記憶領域、第2パラメータ記憶領域および子音アドレス記憶領域に記憶される。また、この編集処理を行う動作モードでは、波形データおよびパラメータ等の、図1に示すハードディスク50への記憶やハードディスク50から波形メモリ20への転送も行われる。この編集処理を行う動作モードも本発明の主題には直接は関係しない。このため、編集処理を行う動作モードに関するこれ以上の説明は省略し、以下の説明では、図3～図5に示す、第1パラメータ記憶領域、第2

パラメータ記憶領域および子音アドレス記憶領域に記憶されるパラメータ等が既に編集作成されているものとする。

【0035】再生処理を行う動作モードについては以下で詳述する。ここで、上述した各種の設定や、モードの選択を行うための操作子群60について図6を参照しながら説明する。操作子群60には、上述した動作モードを設定するスイッチMODE\_SW601と、上述した発音モードを設定するスイッチASSIGN\_SW602が含まれ、また、再生処理を終了するためのスイッチEXIT611が設けられている。

【0036】ボイスが発音されているときにスイッチASSIGN\_SW602が操作されて発音モードが変更されると、ボイスを消音する処理手順が変更されることから、発音中のボイスが止まらなくなるという不都合が起こり得る。この不都合を防止するため、ボイスが発音されているときにスイッチASSIGN\_SWが操作された場合には、一旦、発音中のボイスが全て消音され、押鍵中の鍵に対応するノート情報に従って新たな発音モードで発音が再開される。

【0037】操作子群60には、更に、波形メモリ20の波形データ記憶領域(図2参照)に記憶されている複数の波形データから1つの波形データを原波形の波形データとして選択するためのスイッチBANK\_SW603と、上述したフォルマント比率の設定を行う操作子FORMANT\_VR604と、上述した圧縮伸長率の設定を行う操作子TCOMP\_VR605が設けられている。

【0038】また、操作子群60には、ピッチ変化率、フォルマント比率、音量情報および圧縮伸長率それぞれについて、上述した自動周期変化の程度を設定するための各操作子M P I T C H \_ V R 6 0 6 , M F O R M A N T \_ V R 6 0 7 , M L E V E L \_ V R 6 0 8 , M T C O

M P \_ V R 6 0 9 が設けられている。操作子M P I T C H \_ V R 6 0 6 と操作子M F O R M A N T \_ V R 6 0 7 での設定はセント単位で設定することができる。ここでセントとは1200セントで1オクターブに相当する音程の単位である。操作子M P I T C H \_ V R 6 0 6 , M F O R M A N T \_ V R 6 0 7 , M L E V E L \_ V R 6 0 8 , M T C O M P \_ V R 6 0 9 によって、以下の範囲内で、自動周期変化の程度を設定することができる。

【0039】

$0 \leq$  ピッチ変化率の自動周期変化の程度  $\leq 1200$   
 $0 \leq$  フォルマント比率の自動周期変化の程度  $\leq 1200$   
 $0 \leq$  音量情報の自動周期変化の程度  $\leq 1$   
 $0 \leq$  圧縮伸長率の自動周期変化の程度  $\leq 1$

また、操作子群60には、表示パネル610が設けられており、上記スイッチ等による設定値等が表示される。

【0040】上述した、録音処理を行う動作モードおよび編集処理を行う動作モードにおいて使用される操作子等については図示を省略している。以下、本実施形態の波形生成装置100の動作を、波形図、C P U部80用のプログラムのフローチャート、およびD S P部30用のプログラムのフローチャートを参照して詳細に説明するが、D S P部30のワーキングメモリ、およびC P U部80のワーキングメモリ81が多数のレジスタやカウンタや変数として使用されるので、説明に先立ってこれらのレジスタや変数やカウンタについての一覧表を掲げ説明を付す。なお、一覧表に掲げられたレジスタ等のうちでレジスタ等の名前の最後に「(n)」が付されているレジスタ等は、2つのボイスモジュールを区別するための変数nの値によって選択される、2つのボイスモジュールそれぞれに対応づけられた2つのレジスタ等を意味している。

【0041】

【表1】

レジスタ	記憶している変数等
VOICE_NO.	ボイスモジュールの番号(1または2)
BANK	波形記憶領域に記憶されている波形の番号
ORG. PITCH	基準ピッチを表す変数org. pitch
FORMANT	フォルマント比率を表す変数formant
T COMP	圧縮伸長率を表す変数t comp
M PITCH	ピッチの自動周期変化の程度を表す変数mpitch
M FORMANT	フォルマントの自動周期変化の程度を表す変数mformant
M LEVEL	音量の自動周期変化の程度を表す変数mlevel
M T COMP	圧縮伸長率の自動周期変化の程度を表す変数mtcomp
PITCH(n)	音高情報note_no.に対応するピッチを表わす変数note_pitchを基準ピッチを表す変数org. pitchで割った値を記憶
LEVEL(n)	ペロシティ情報velocity
LFO_GATE(n)	自動周期変化の発音開始時には値「1.0」、自動周期変化の発音停止時には値「0.0」を記憶
NOTE_No.	音高情報note_no.
VELOCITY	ペロシティ情報velocity
ON/OFF	オンオフ情報on/off

【0042】

【表2】

フラグレジスタ	フラグで表わされる内容
AS_FLG	割当処理によるボイスモジュールへの割当が1ボイスならば「MONO」複数ボイスならば「POLY」
LINK_MODE	発音モードがMONO2またはPOLY2であれば「1」、MONO1またはPOLY1であれば「0」
Link(n)	リンク発音を行うならば「1」、行わないならば「0」
F(n)	第1チャンネルを表すときは「1」 第2チャンネルを表すときは「-1」
Vflag(n)	原波形の子音部に基づいて出力波形を形成中ならば「1」 原波形の母音部に基づいて出力波形を形成中ならば「0」

【0043】

【表3】

D S P の変数	内 容
Bank (n)	波形領域に記憶されている波形のうちの、原波形として選択された波形の番号
Pitch (n)	ピッチの変化率
Formant (n)	フォルマント比率
Level (n)	音量情報
Tcomp (n)	圧縮伸長率
Width (n)	出力波形のピッチに相当する時間幅をサンプリング周期を単位として表わす変数
Length (n)	図13 (d), (e) に示す三角波の底辺の半分の長さ
Endadrs (n)	波形データの最終アドレス
Gate (n)	発音開始時に「1. 0」、発音停止時に「0. 0」となる
Out (n)	出力波形の波形データ
Out	各ボイスモジュールにより形成された各出力波形の波形データの和
Sphase (n)	変数Tcomp (n) が示す圧縮伸長率に従う速度で原波形上を移動する再生基準点の位置を示す、波形メモリ20のアドレス
Spitch (n)	原波形の区分のうちの、変数Sphase (n) で示される再生基準点を含む区分のピッチ
Start (n)	原波形の区分のうちの、変数Sphase (n) で示される再生基準点を含む区分の先頭アドレス
Sadrs (n)	原波形の区分のうちの、変数Sphase (n) で示される再生基準点を含む区分の次の区分の先頭アドレス
Vstart (n)	子音部の先頭アドレス
Vend (n)	子音部の末尾のアドレス
Start1 (n)	原波形の各区分の各先頭アドレスのうちの原波形を第1チャンネル用に読み込むための基準となる先頭アドレス
Start2 (n)	原波形の各区分の各先頭アドレスのうちの原波形を第2チャンネル用に読み込むための基準となる先頭アドレス
W_rate (n)	関数Window1および関数Window2の増加率
Adres1	原波形の波形データを波形メモリから第1チャンネル用に読み込むためのアドレス
Adres2	原波形の波形データを波形メモリから第2チャンネル用に読み込むためのアドレス

【0044】

【表4】

カウンタ	カウントしているもの
Scnt (n)	原波形の先頭から数えた区分数
Ph1 (n)	変数Sphase (n) で示される再生基準点の位置のアドレスが、変数Start1 (n) で示されるアドレスと同一であった時点から数えたサンプリング周期数
Ph2 (n)	変数Sphase (n) で示される再生基準点の位置のアドレスが、変数Start2 (n) で示されるアドレスと同一であった時点から数えたサンプリング周期数
Vcnt (n)	原波形の先頭から数えた子音部数

【0045】ここで、フォルマント比率を表す変数formantの値が「1.0」ならば、出力波形のフォルマントは原波形のフォルマントと同じフォルマントとなり、変数formantの値が「1.0」より大きい場合は、出力波形のフォルマントは原波形のフォルマントより高音域側に遷移し、「1.0」より小さい場合は、出力波形のフォルマントは原波形のフォルマントより低音域側に遷移する。

【0046】また、圧縮伸長率を表す変数tcompの値が「1.0」ならば原波形全体が発音されるのに要する時間と同じ時間で出力波形全体が発音される。変数tcompの値が「1.0」より大きい場合は、原波形全体が発音されるのに要する時間より短い時間で出力波形全体が発音され、原波形が例えば言葉を表す波形ならば早口言葉になる。変数tcompの値が「1.0」より小さい場合は、原波形全体が発音されるのに要する時間より長い時間で出力波形全体が発音される。

【0047】以下では、一覧表に掲げたレジスタ等について、特に説明を付さずに使用する場合がある。また、各レジスタの値を用いた計算等を説明する場合には式の中で、変数と、その変数を記憶するレジスタとを区別せずに用いる場合がある。図7はCPU部用のプログラムのメインルーチンのフローチャートである。このメインルーチンでは、図1に示すCPU部80の、上述した3つの動作モードのうちの1つの動作モードが選択される。

【0048】メインルーチンが起動されると、まず、動作モードを選択するため、図6に示すスイッチMODE\_SW601の検出が行われ、スイッチMODE\_SW601が操作されたか否かがステップS201において判定される。スイッチMODE\_SW601が操作されるまでの間はスイッチMODE\_SW601の検出と判定が繰り返され、スイッチMODE\_SW601が操作されたと判定されるとステップS202に進む。ステップS202では、スイッチMODE\_SW601によって選択された動作モードが判定され、録音モード(REC)であると判定された場合にはステップS203に

進み、編集モード(EDIT)であると判定された場合にはステップS204に進み、再生モード(PLAY)であると判定された場合にはステップS205に進む。

【0049】録音モードおよび編集モードの動作についてはこれまでに説明した通りであり、また、上述したように、録音モードおよび編集モードの詳細な内容は本発明の主題ではないので説明を省略する。ステップS205では後述する再生処理が行われる。ステップS203～ステップS205の各処理が終わるとS201に戻り上記の過程が繰り返される。

【0050】図8は、図7のステップS205に示す再生処理ルーチンのフローチャートである。この再生処理ルーチンが起動されると、ステップS401で、図6に示す操作子群60の各スイッチ等それぞれによって設定される各設定値に対して初期設定が行われ、その初期設定に合わせて、表1および表2に示す各レジスタの値が初期化される。その後、ステップS402に進んで、図6に示す、再生処理を終了させるためのスイッチEXIT611が押されているか否かが判定される。スイッチEXIT611が押されていると判定されるとメインルーチンに戻り、スイッチEXIT611が押されていないと判定されるとステップS403に進んで、後述する、再生処理の主要部分の処理が行われる。ステップS403が終わるとステップS402に戻る。

【0051】図9は、図8のステップS403に示すルーチンの前半部分のフローチャートである。図10は、図8のステップS403に示すルーチンの後半部分のフローチャートであり、図9のステップS613およびステップS614から、図10のステップS615へと処理の流れが続く。

【0052】このルーチンが起動されると、ステップS601において、図6に示す、波形データを選択するスイッチBANK\_SW603が操作されているか否かが判定され、操作されていると判定されるとステップS602に進んで、レジスタBANKに記憶されている波形の番号が、スイッチBANK\_SW603の操作に応じて更新され、ステップS603に進んでレジスタBAN

Kに記憶されている波形の番号に対応する波形データの基準ピッチが、図1に示す波形メモリ20の第2パラメータ記憶領域(図4参照)からDSP部30を経由してCPU部80に読み込まれてレジスタORG.PITCHに記憶された後、ステップS604に進む。ステップS601においてBANK\_SW603が操作されないと判定された場合は、何もせずにステップS604に進む。

【0053】ステップS604では、図6に示す、フォルマント比率を設定する操作子FORMANT\_VR604が操作されているか否かが判定され、操作されないと判定されるとステップS605に進んで、レジスタFORMANTに記憶されている変数formantの値が、操作子FORMANT\_VR604の操作に応じて更新され、ステップS606に進む。ステップS604において、操作子FORMANT\_VR604が操作されないと判定された場合は何もせずにステップS606に進む。

【0054】ステップS606では、図6に示す、圧縮伸長率を設定する操作子TCOMP\_VR605が操作されているか否かが判定され、操作されないと判定されるとステップS607に進んで、レジスタTCOMPに記憶されている変数tcompの値が、操作子TCOMP\_VR605の操作に応じて更新され、ステップS608に進む。ステップS606において、操作子TCOMP\_VR605が操作されないと判定された場合は何もせずにステップS608に進む。

【0055】ステップS608では、図6に示す、発音モードを選択するスイッチASSIGN\_SW602に変化が有るか否かが判定され、スイッチASSIGN\_SW602に変化が有ると判定されるとステップS609に進む。ステップS609において、スイッチASSIGN\_SW602によって設定された発音モードが、レガート演奏の際に原波形の途中からボイスの発音を行う、MONO2モードおよびPOLY2モードのいずれかの発音モードであると判定されるとステップS610でフラグレジスタLINK\_MODEが「1」に設定されてステップS612に進み、スイッチASSIGN\_SW602によって設定された発音モードが、レガート演奏の際に原波形の先頭からボイスの発音を行う、MONO1モードおよびPOLY1モードのいずれかの発音モードであると判定されるとステップS611でフラグレジスタLINK\_MODEが「0」に設定されてステップS612に進む。

【0056】ステップS612において、スイッチASSIGN\_SW602によって設定された発音モードが、1つの出力波形を形成する、MONO1モードおよびMONO2モードのいずれかの発音モードであると判定されるとステップS613でフラグレジスタAS\_FLGが「MONO」に設定されてステップS615に進

み、スイッチASSIGN\_SW602によって設定された発音モードが、レガート演奏時に2つの出力波形を形成する、POLY1モードおよびPOLY2モードのいずれかの発音モードであると判定されるとステップS614でフラグレジスタAS\_FLGが「POLY」に設定されてステップS615に進む。

【0057】ステップS608において、スイッチASSIGN\_SW602に変化がないと判定された場合には、なにもせずにステップS615に進む。ステップS615では、図6に示す、ピッチの自動周期変化の程度を設定する操作子MPITCH\_VR606が操作されているか否かが判定される。操作されていると判定されるとステップS616に進んでレジスタMPITCHに記憶されている変数pitchの値が、操作子MPITCH\_VR606の操作に応じて更新されてステップS617に進み、操作されていないと判定されると何もせずにステップS617に進む。

【0058】同様に、ステップS617, S619およびS621それぞれでは、図6に示す、操作子MFORMANT\_VR607、操作子MLEVEL\_VR608および操作子MTCOMP\_VR609それが操作されているか否かが判定され、操作されていると判定された場合には、ステップS618、S620およびS622に進んでレジスタMFORMANT、レジスタMLEVELおよびレジスタMTCOMPそれぞれに記憶されている変数formant、変数levelおよび変数tcompそれぞれの値が各操作子の操作に応じて更新される。

【0059】その後ステップS623に進み、鍵盤70からノート情報が入力されていると判定されると、ノート情報がレジスタNOTE\_No.、レジスタVELOCITYおよびレジスタON/OFFに記憶され、ステップS624において、後述する、ボイスモジュールへボイスを割当てる割当処理が行われる。その後ステップS625に進み、後述する、DSPへ各種のパラメータを転送する転送処理が行われて図8に示す再生処理ルーチンへ戻る。

【0060】図11は、ボイスモジュールへの割当処理ルーチンのフローチャートである。このルーチンが起動され、ステップS801において、フラグレジスタAS\_FLGの設定が「MONO」であると判定されると、ステップS802に進み、1つのボイスモジュールに対するボイスの割当処理が、いわゆるジェネレーターアシンの技術を用いて行われ、ボイスが割り当てられたボイスモジュールの番号がレジスタVOICE\_NO.に記憶されてステップS804に進む。ステップS801においてフラグレジスタAS\_FLGの設定が「POLY」であると判定されると、ステップS803に進み、2つのボイスモジュールに対するボイスの割当処理が行われ、ボイスが割り当てられたボイスモジュールの番号

がレジスタVOICE\_NO. に記憶されてステップS 804に進む。

【0061】ステップS 804では変数nにレジスタVOICE\_NO. の値が代入される。以後、変数nの値に応じてボイスモジュールの使い分けが行われる。その後、ステップS 805に進み、レジスタON/OFFに記憶されているオンオフ情報on/offが離鍵を表していると判定されると、ステップS 806に進みレジスタLEVEL(n)の値が「0」に設定されることによって音量がゼロになり、ステップS 807に進みレジスタLFO\_GATE(n)の値が「0.0」に設定されることによってピッチ等の自動的な周期変化の程度がゼロになった後、図10に示すルーチンに戻る。

【0062】ステップS 805においてオンオフ情報on/offが押鍵を表していると判定されると、ステップS 808～ステップS 814において、以下述べるような、ボイスの発音状態の設定が行われる。まず、ステップS 808において、レジスタNOTE\_NO. に記憶されている音高情報note\_no. が表す音高に対応するピッチを表す変数note\_pitchの値が求められる。次にステップS 809に進み、レジスタPITCH(n)が、変数note\_pitchの値を変数org\_pitchの値で割った値に設定され、レジスタLEVEL(n)が、レジスタVELOCITYに記憶されているペロシティ情報velocityの値に設定される。これにより、出力波形のピッチおよび音量の基準が設定される。また、レジスタLFO\_GATE(n)が値「1.0」に設定され、これによりピッチ等の自動的な周期変化が開始される。

【0063】ステップS 810において、フラグレジスタLINK\_MODEの値が「1」であり、即ちレガート演奏の際に原波形の途中から発音する発音モードであると判定され、かつステップS 811において、レガート演奏が行われていると判定されると、ステップS 812に進む。ステップS 812において、フラグレジスタAS\_FLGが「POLY」を表していると判定されると、ステップS 813で、DSP部のフラグレジスタlink(n)が、リンク発音が行われることを示す値「1」に設定されて、図10に示すルーチンに戻り、フラグレジスタAS\_FLGが「MONO」を表していると判定されると、そのまま、図10に示すルーチンに戻る。

【0064】ステップS 810において、フラグレジスタLINK\_MODEの値が「0」であると判定されたかあるいは、ステップS 811において、レガート演奏が行われていないと判定された場合にはステップS 814に進み、DSP部の変数Bank(n)にレジスタBANKの値が代入されることにより、出力波形の形成に用いられる原波形が選択され、DSP部の、原波形の区分数を示すカウンタScnt(n)の値が「0」にリセ

ットされることにより、以下で説明するように、変数nで示されるボイスの発音が開始される。その後、図10に示すルーチンに戻る。

【0065】図12はDSPへの転送処理ルーチンのフローチャートである。このDSPへの転送処理ルーチンは、図9および図10に示す、再生処理の主要部分を表すルーチンが起動される度に、ステップS 625において起動されるルーチンである。このDSPへの転送処理ルーチンでは、図1に示すDSP部30で用いられる変数Pitch(n)、変数Formant(n)、変数Level(n)および、変数Tcomp(n)の値が求められてDSP部30へ転送される。

【0066】このDSPへの転送処理ルーチンが起動されると、ステップS 1001において、ステップS 1001を通過する度にレジスタLFO\_GATE(n)の設定値に順次近づく関数LFO\_ENV(n)の値が更新される。即ち、レジスタLFO\_GATE(n)の設定値から現在の関数LFO\_ENV(n)の値を引いた結果と係数SLOPEとの積が現在の関数LFO\_ENV(n)の値に加えられて関数LFO\_ENV(n)の新しい値となる。なお、フローチャートでは、記号「+」によって、左辺の変数やレジスタの値が、右辺の式の値だけ累増することが示されている。また、自動周期変化の速さを決める係数LFO\_RATE、およびステップS 1001を通過する度に歩進されるカウンタLFO\_COUNTの値に基づいて、次式に示すように、ピッチ等を自動的に周期変化させるための周期関数LFOの値が更新される。

【0067】 $LFO = \sin(2 * 3.14 * LFO\_RATE / 32000 * LFO\_COUNT)$

その後、カウンタLFO\_COUNTの値が歩進されてステップS 1002に進む。なお、フローチャートでは、記号「++」によって、カウンタや変数等が歩進されることが示されている。

【0068】ステップS 1002では、変数Pitch(n)、変数Formant(n)、変数Level(n)および、変数Tcomp(n)の値が次の各式で計算される。

$$\begin{aligned} Pitch(n) &= PITCH(n) * \text{POW}(0.5, LFO * LFO\_ENV(n) * M_PITCH / 1200) \\ Formant(n) &= FORMANT * \text{POW}(0.5, LFO * LFO\_ENV(n) * M_FORMANT / 1200) \end{aligned}$$

$$\begin{aligned} Level(n) &= LEVEL(n) * 0.5 * (1 + LFO * LFO\_ENV(n) * M_LEVEL) \\ Tcomp(n) &= TCOMP * \text{POW}(0.5, LFO * LFO\_ENV(n) * M_TCMP) \end{aligned}$$

ここで、関数POW(X, Y)とはXのY乗を表す関数であり、上述したように、変数とレジスタとを区別せず

に用いている。また、上述したように、各レジスタの値は以下の範囲内の値に設定されている。

【0069】 $0 \leq M P I T C H \leq 1200$

$0 \leq M F O R M A N T \leq 1200$

$0 \leq M L E V E L \leq 1$

$0 \leq M T C O M P \leq 1$

なお、変数 *Formant* (*n*) および変数 *T comp* (*n*) については、上式では、2つのボイスモジュールに共通するレジスタ *FORMANT* およびレジスタ *T COMP* の値が用いられて計算されているが、2つのボイスモジュールそれぞれに対応付けられたレジスタが設けられ、それらのレジスタの値が、レジスタ *FORMANT* およびレジスタ *T COMP* の値に代えて用いられて上記の式の計算が行われても良い。

【0070】その後ステップ S 1003 に進み、変数 *Pitch* (*n*)、変数 *Formant* (*n*)、変数 *Level* (*n*) および、変数 *T comp* (*n*) が、図 1 に示す CPU 部 80 から DSP 部 30 へ転送された後、図 10 に示すルーチンヘリターンする。以下 DSP 部 30 が出力波形を形成する動作について説明するが、先ず図 13 の波形図を参照して、1つのボイスに対応する1つの出力波形が形成される動作を概説し、その後フローチャートおよび波形図を参照して DSP 部 30 の動作の詳細な内容を説明する。

【0071】図 13 (a) には、図 1 に示す波形メモリ 20 に記憶されている波形データが表す原波形の一例が示されており、この原波形には非周期的な振る舞いをする子音部と周期的な振る舞いをする母音部とが含まれている。また、この波形図には、波形データの区分 *spitcho* の先頭から区分 *spitcho* 5 の途中までが表されており、区分 *spitcho* 0 から区分 *spitcho* 2 までが子音部であり、区分 *spitcho* 3 以降が母音部である。

【0072】この原波形から出力波形を形成する方法として本実施形態では、便宜的に2つのチャネルを用意し、それらの2つのチャネルそれぞれに対応付けられた2つの波形を一旦形成し、それらの2つの波形を足しあわせて出力波形を形成する方法が用いられる。その手順を概括的に説明すると、まず、変数 *Width* の値と同じ値の高さを有する鋸歯状波を表す関数 *Phase* を作り（図 13 (c)）、関数 *Phase* に同期させて、2つのチャネルそれぞれに対応する2つの三角波列それそれを表す関数 *Env1* および関数 *Env2* を、各三角波の底辺の長さが変数 *Length* の値の2倍となるように作る（図 13 (d), (e) の上段）。次に、圧縮伸長率に基づいて、関数 *Env1* および関数 *Env2* が表す各三角波それぞれに対して波形データの区分を対応付け、その対応付けられた各波形データの区分から、フォルマント比率に従うアドレス間隔で、各チャネル用のデータを順次読み込む（図 13 (d), (e) の下

段）。なお、図 13 (d), (e) の下段には、図 13 (a) に示す波形データの各区分が、波形が省略されて示されている。その後、読み込まれた各チャネル用のデータと、関数 *Env1* および関数 *Env2* それぞれの値とを掛け合わせて各チャネル用の波形を表す各データを形成する（図 13 (f), (g)）。最後に、このようにして形成された、2つの波形を表す2つのデータを足しあわせて出力波形を表す波形データを形成する。

【0073】原波形の母音部に基づいて出力波形が形成される場合には、図 12 のステップ S 1002 において計算された変数 *Pitch* (*n*), *Formant* (*n*), *T comp* (*n*) の値が示すピッチ変化率等に応じて、関数 *Phase* の周期や、関数 *Env1* および関数 *Env2* の三角波の底辺の長さ等が決定されて出力波形が形成される。図 13 には、原波形の母音部に基づいて、値が「1.0」である圧縮伸長率と、「1.0」より小さい値であるピッチ変化率と、「1.0」より大きい値であるフォルマント比率に応じて出力波形のうちの母音部に対応する部分が形成される過程が示されている。出力波形のうちの、母音部に対応する部分の波形は、原波形の母音部の波形に対してピッチおよびフォルマントがいずれも高音域側に変化した波形となっている。母音部は上述のように周期的な振る舞いをしているので、このようにピッチ等が変換されても出力される音は自然に聞こえる。

【0074】一方、原波形の子音部に基づいて出力波形が形成される場合には、図 12 のステップ S 1002 において計算された変数 *Pitch* (*n*), *Formant* (*n*), *T comp* (*n*) の計算値が示すピッチ変化率等によらず、ピッチ変化率、圧縮伸長率およびフォルマント比率それぞれが常に「1.0」と設定されたかのように出力波形が形成される。その結果、出力波形のうちの、子音部に対応する部分の波形は、原波形の子音部の波形と同一の波形となり、出力される音は自然な音色となる。

【0075】なお、子音部に基づいて出力波形が形成される場合のピッチ変化率等は必ずしも「1.0」である必要はなく、音色が不自然にならない程度のピッチ変化率等に従って子音部の波形が変換されても良い。以下では、子音部に基づいて出力波形が形成される場合のピッチ変化率等の値はすべて「1.0」と設定されたかのように出力波形が形成されるものとして説明する。

【0076】以下フローチャートおよび波形図を参照しながら図 1 に示す DSP 部 30 の動作について詳しく説明する。図 14 は、DSP 部用のプログラムのメインループのフローチャートである。このメインループは、出力波形信号のサンプリング周期毎に起動され、図 13 を用いて上記説明した動作に入る前に、出力波形を形成するための各種の変数値やレジスタの値等を設定したり、それらの値を一方のボイスモジュールから他方の

ボイスモジュールへ受渡したりして、その後、図13を用いて上記説明した動作に入る。

【0077】また、フローチャートには記載されていないが電源投入時には、カウンタS<sub>c nt</sub>(n)、変数W<sub>i d t h</sub>(n)、変数P<sub>i t c h</sub>(n)、変数L<sub>e v e l</sub>(n)および変数E<sub>n d a d r s</sub>(n)のそれぞれの値が「0」に初期設定され、チャンネルを表すフラグレジスタF(n)の値が第1のチャンネルを表す値である「1」に初期設定される。なお、第2のチャンネルを表す値は「-1」である。また、圧縮伸長率を表す変数T<sub>c o m p</sub>(n)の値が「1.0」に初期設定され、図13(d), (e)に示す関数W<sub>i n d o w 1</sub>(n)および関数W<sub>i n d o w 2</sub>(n)のそれぞれの値が「0.0」および「1.0」に初期設定される。また、フラグレジスタL<sub>i n k</sub>(n)が、リンク発音が行われないことを表す値「0」に初期設定され、変数G<sub>a t e</sub>(n)が「0.0」に初期設定され、カウンタV<sub>c nt</sub>(n)の値が「0」に初期設定され、フラグレジスタV<sub>f l a g</sub>(n)の値が、原波形の母音部に基づいて出力波形を形成中であることを示す値である「0」に初期設定される。なお、原波形の子音部に基づいて出力波形を形成中であることを示す値は「1」である。

【0078】このメインルーチンのステップS1201～ステップS1208では、リンク発音用にボイスモジュールの設定値の受渡しが行われるリンク発音処理、およびボイスモジュールによる発音開始用にボイスモジュールの設定値の初期化が行われる発音開始処理が行われる。ステップS1201において、フラグレジスタL<sub>i n k</sub>(1)の値が、リンク発音が行われることを示す値「1」であると判定されると、ステップS1202に進んで、後述する、2つのボイスのうちの第2のボイスから第1のボイスへのリンク発音用のリンク発音処理が行われることにより、第1のボイスが、第2のボイスの発音中の音に対応する原波形上の位置と同じ位置に対応する音から発音される。その後、ステップS1203に進む。

【0079】ステップS1201において、フラグレジスタL<sub>i n k</sub>(1)の値が、リンク発音が行われないことを示す値「0」であると判定された場合は、なにもせずにステップS1203に進む。同様に、ステップS1203において、フラグレジスタL<sub>i n k</sub>(2)が「1」であると判定されると、ステップS1204に進んで、2つのボイスのうちの第1のボイスから第2のボイスへのリンク発音用のリンク発音処理が行われ、ステップS1205に進む。ステップS1203において、フラグレジスタL<sub>i n k</sub>(2)が「0」であると判定された場合は、なにもせずにステップS1205に進む。

【0080】ステップS1205において、カウンタS<sub>c nt</sub>(1)の値が「0」にリセットされていると判定されると、ステップS1206に進んで第1のボイスの

発音開始処理が行われる。これによって、第1のボイスの発音が開始される。その後、ステップS1207に進む。ステップS1205において、カウンタS<sub>c nt</sub>

(1)の値が「0」にリセットされていないと判定された場合は、なにもせずにステップS1207に進む。

【0081】同様に、ステップS1207において、カウンタS<sub>c nt</sub>(2)の値が「0」にリセットされていると判定されると、ステップS1208に進んで第2のボイスの発音開始処理が行われ、ステップS1209に進む。ステップS1207において、カウンタS<sub>c nt</sub>(2)の値が「0」にリセットされていないと判定された場合は、なにもせずにステップS1209に進む。

【0082】その後、ステップS1209～ステップS1212において、第1および第2のボイスそれについて、上記で図13を用いて説明した動作を行う読み出し処理が行われる。ただし、フローチャートでは、記号「>=」および記号「<=」が用いられて、以上・以下の関係が表されている。読み出し処理の詳細は後述する。その後、ステップS1213に進み、第1のボイスに対応する出力波形を表す変数O<sub>u t</sub>(1)の値と第2のボイスに対応する出力波形を表す変数O<sub>u t</sub>(2)の値が足し合わせて変数O<sub>u t</sub>の値が求められ、ステップS1214に進み、変数O<sub>u t</sub>の値が 출력される。

【0083】図15は、第2のボイスから第1のボイスへのリンク発音処理ルーチンのフローチャートである。このルーチンは、図14に示す、DSPのメインルーチンのステップS1201において、フラグレジスタL<sub>i n k</sub>(1)の値が、リンク発音が行われることを示す値「1」であると判定されると、ステップS1202において起動されるルーチンである。

【0084】このルーチンでは、第2のボイスから第1のボイスへのリンク発音が行われるように、以下に挙げる第2のボイス用の変数、カウンタおよびレジスタそれぞれの値がそのまま第1のボイス用の変数、カウンタおよびレジスタそれぞれに代入される。このルーチンにおいて第2のボイス側から第1のボイス側へ値がそのまま受け渡される変数、カウンタおよびレジスタは、変数B<sub>a n k</sub>(2)、変数T<sub>c o m p</sub>(2)、カウンタS<sub>c nt</sub>(2)、変数S<sub>p i t c h</sub>(2)、変数S<sub>a d r s</sub>(2)、変数E<sub>n d a d r s</sub>(2)、カウンタP<sub>h 1</sub>(2)、カウンタP<sub>h 2</sub>(2)、カウンタV<sub>c nt</sub>(2)、フラグレジスタV<sub>f l a g</sub>(2)、変数V<sub>s t a r t</sub>(2)および変数V<sub>e n d</sub>(2)である。

【0085】また、このルーチンでは、変数S<sub>p h a s e</sub>(1)、変数S<sub>t a r t</sub>(1)、変数S<sub>t a r t</sub>1(1)および変数S<sub>t a r t</sub>2(1)のすべてに変数S<sub>p h a s e</sub>(2)の値が代入され、これによって、原波形の、第2のボイス用に波形データが読み込まれている区分と同じ区分から第1のボイス用に波形データが読み込まれることとなる。

【0086】また、変数Gate(1)が「1.0」に設定されることにより第1のボイスの発音が開始され、フラグレジスタLink(1)が「0」に設定されることによりこのルーチンの不要な再起動が行われなくなる。上記の代入および設定を終了した後、DSPのメインルーチンへ戻る。図16は、第1のボイスから第2のボイスへのリンク発音処理ルーチンのフローチャートである。

【0087】このルーチンは、図14に示す、DSPのメインルーチンのステップS1203において、フラグレジスタLink(2)の値が、リンク発音が行われることを示す値「1」であると判定されると、ステップS1204において起動されるルーチンである。このルーチンでは、第1のボイスから第2のボイスへのリンク発音が行われるように、図15に示すルーチンと同様の代入および設定が行われる。説明は重複するので省略する。

【0088】図17は、第1のボイスの発音開始処理ルーチンのフローチャートである。このルーチンは、図14に示す、DSPのメインルーチンのステップS1205において、カウンタScnt(1)の値が「0」にリセットされていると判定されると、ステップS1206において起動されるルーチンである。このルーチンでは、第1のボイスの発音開始用に第1のボイスモジュールの設定値の初期化が行われる。

【0089】まず、波形メモリの第1パラメータ記憶領域(図3参照)の、変数Bank(1)の値に基づく位置から、波形データの末尾のアドレスが読み込まれ変数Endadr(1)に代入される。なお、フローチャートでは@(X)という表現を用いて、Xで示される位置から読み込まれた値を表現している。同様に、波形データの先頭のアドレスが読み込まれて変数Phase(1)に代入され、波形データの区分spitch0のピッチが読み込まれて変数Spitch(1)に代入され、波形データの区分spitch1の先頭のアドレスが読み込まれて変数Sadrs(1)に代入される。

【0090】このルーチンでは、変数Start(1)、変数Start1(1)および変数Start2(1)のすべてに変数Phase(1)の値が代入される。これによって、波形の先頭から波形データの読み込みが行われることとなる。変数Width(1)には、変数Spitch(1)が示すピッチと、変数Pitich(1)が示すピッチ変化率との積が代入され、これにより、出力波形のピッチの初期値が設定される。

【0091】カウンタPh1(1)の値は「0」に初期設定され、カウンタPh2(1)の値は「Width(1)/2」に初期設定される。図13(d), (e)に示す鋸歯状波を表す関数Window1(1)および関数Window2(1)は、それぞれ「0.0」および「1.0」に初期設定され、これにより図13

(d), (e)に示すように、関数Window1(1)と関数Window2(1)とは互いに180度位相が異なった関数となる。

【0092】変数Gate(1)は「1.0」に設定され、これにより、第1のボイスの発音が開始される。カウンタScnt(1)の値は「1」に初期設定され、カウンタVcnt(1)の値は「0」に初期設定され、フラグレジスタVflag(1)の値は「0」に初期設定される。このルーチンでは、波形メモリ20の子音アドレス記憶領域(図5参照)の、変数Bank(1)の値に基づく位置から、波形データの最初の子音部の子音開始アドレスと、波形データの最初の子音部の子音終了アドレスとが読み込まれ、それぞれ変数Vstart(1)と変数Vend(1)に代入される。

【0093】上記の初期設定が終了した後、DSPのメインルーチンに戻る。図18は、第2のボイスの発音処理ルーチンのフローチャートである。このルーチンは、図14に示す、DSPのメインルーチンのステップS1207において、カウンタScnt(2)の値が「0」にリセットされていると判定されると、ステップS1208において起動されるルーチンである。

【0094】このルーチンでは、図17に示す第1のボイスの発音処理ルーチンと同様に、第2のボイスの発音開始用に第2のボイスモジュールの設定値の初期化が行われる。説明は重複するので省略する。図19は、読み出し処理ルーチンの前半部分を示すフローチャートである。図20は、読み出し処理ルーチンの後半部分を示すフローチャートであり、図19に示すステップS1412から図20に示すステップS1413へと処理の流れが続く。

【0095】このルーチンは、図14に示す、DSPのメインルーチンが起動される度に、ステップS1210において起動されるルーチンである。この読み出し処理ルーチンのステップS1401～ステップS1412においては、圧縮伸長率に従う速度で原波形上を移動する再生基準点の位置の判定が行われたり、その判定結果に基づいて、波形メモリ20からパラメータが読み込まれたりする。

【0096】読み出し処理ルーチンが起動されると、まず、ステップS1401～ステップS1405において、変数Phase(n)が示す再生基準点と、子音部との位置関係に基づく処理が行われる(図13(b)参照)。ステップS1401において、変数Phase(n)の値が変数Vstart(n)の値よりも小さいと判定され、即ち、再生基準点が子音部の手前に位置していると判定されると、ステップS1405に進んでフラグレジスタVflag(n)の値が「0」に設定される。

【0097】また、ステップS1401およびステップS1402において、変数Phase(n)の値が、変数Vstart(n)の値以上であって、かつ、変数

*Vend* (n) の値以下であると判定され、即ち、再生基準点が子音部内に位置していると判定されると、ステップ S1403 に進み子音フラグ *Vflag* (n) が「1」に設定される。

【0098】また、ステップ S1401 およびステップ S1402において、変数 *Phase* (n) の値が、変数 *Vstart* (n) の値以上であって、かつ、変数 *Vend* (n) の値よりも大きいと判定され、即ち、再生基準点が子音部を通り越した位置に存在すると判定されると、ステップ S1404 に進んで、変数 *Vstart* (n) および変数 *Vend* (n) それぞれの値が、1つ先の子音部の子音開始アドレスおよび子音終了アドレスそれぞれの値に更新される。すなわち、カウンタ *Vcnt* (n) が歩進され、波形メモリ 20 の子音アドレス記憶領域（図 5 参照）の、変数 *Bank* (n) の値とカウンタ *Vcnt* (n) の値に基づく位置から、カウンタ *Vcnt* (n) が示す子音部の子音開始アドレスおよび子音終了アドレスが読み込まれ、それぞれ変数 *Vstart* (n) と変数 *Vend* (n) に代入される。

【0099】その後ステップ S1405 に進み子音フラグ *Vflag* (n) が「0」に設定される。その後、ステップ S1406～ステップ S1408において、変数 *Phase* (n) の値が累増され、これにより、再生基準点が原波形上を移動する。再生基準点が子音部内である場合には、変数 *Phase* (n) の値が累増される大きさは「1.0」であり（ステップ S1408）、再生基準点が子音部外であり即ち母音部内である場合には、変数 *Phase* (n) の値が累増される大きさは圧縮伸長率を表す変数 *Tcomp* (n) の値となる（ステップ S1409）。なお、変数 *Phase* (n) の値が累増されることによって再生基準点が波形の末尾を越えると、対応する波形データが存在しないために不都合を生じるので、ステップ S1409において、変数 *Phase* (n) の値が、波形データの末尾のアドレスを表す変数 *Endadr* (n) の値を越えたと判定されると、ステップ S1410 に進み変数 *Phase* (n) に変数 *Endadr* (n) の値が代入される。

【0100】その後、ステップ S1411 に進み、変数 *Phase* (n) の値が変数 *Sadr* (n) 以上であるか否かが判定され、即ち再生基準点が、原波形上の次の区分に到達したか否かが判定される。到達したと判定されるとステップ S1412 において、変数 *Start* (n) の値に変数 *Sadr* (n) の値が代入され、また、波形メモリの第 1 パラメータ記憶領域（図 3 参照）の、変数 *Bank* (1) の値とカウンタ *Scnt* (n) の値に基づく位置から、カウンタ *Scnt* (n) の値が示す区分のピッチおよび区分先頭アドレスが読み込まれ、それぞれ変数 *Pitch* (n) および変数 *Sadr* (n) に代入される。最後にカウンタ *Scnt* (n) の値が歩進され、その後、ステップ S1413 に

進む。

【0101】ステップ S1411において、再生基準点が、原波形上の次の区分に到達していないと判定された場合は、なにもせずにステップ S1413 に進む。ステップ S1413～ステップ S1424 では、図 13 (c) に示す鋸歯状波の生成や、各種の変数値や関数値の、鋸歯状波に同期した更新等が行われる。まず、ステップ S1413 において、関数 *Phase* (n)、カウンタ *Ph1* (n) およびカウンタ *Ph2* (n) それぞれの値が歩進され、次に、ステップ S1414 において、関数 *Phase* (n) の値が変数 *Width* (n) の値に到達したか否かが判定される。関数 *Phase* (n) の値が変数 *Width* (n) の値に到達していない間は、ステップ S1425 に進み、ステップ S1415～ステップ S1424 は実行されない。これにより、関数 *Phase* (n)、カウンタ *Ph1* (n) およびカウンタ *Ph2* (n) それぞれの値は、関数 *Phase* (n) の値が変数 *Width* (n) の値に到達するまでサンプリング周期毎に歩進され続けることとなる。

【0102】ステップ S1414において、関数 *Phase* (n) の値が変数 *Width* (n) の値に到達したと判定された場合は、ステップ S1415 に進み、関数 *Phase* (n) の値が「0」にリセットされ、これによって関数 *Phase* (n) は、図 13 (c) に示す、変数 *Width* (n) の値と同じ値の高さを有する鋸歯状波を表すこととなる。

【0103】その後、ステップ S1416～ステップ S1420 において、変数 *Width* (n) および変数 *Length* (n) の値がフラグレジスタ *Vflag* (n) の値に応じて更新される。フラグレジスタ *Vflag* (n) の値が、子音部に基づく出力波形の形成を表す値「1」である場合には、変数 *Width* (n) および変数 *Length* (n) の値いずれもピッチ *Pitch* (n) の値に設定され（ステップ S1417）、フラグレジスタ *Vflag* (n) の値が、母音部に基づく出力波形の形成を表す値「0」である場合には、変数 *Width* (n) の値は変数 *Pitch* (n) の値と変数 *Pitch* (n) の値との積の値に設定され、変数 *Length* (n) の値は変数 *Pitch* (n) を変数 *Formant* (n) で割った値に設定される（ステップ S1418）。このため、変数 *Pitch* (n) が「1.0」以外の値に設定されると、図 13 (c) に示すように、子音部に対応する部分と母音部に対応する部分とでは、鋸歯状波の高さが相違することとなり、また、図 13 (d), (e) の上段に示すように、子音部に対応する部分と母音部に対応する部分とでは、三角波の底辺の長さが相違することとなる。また、子音部に対応する部分では、上述したように変数 *Width* (n) の値と変数 *Length* (n) の値は同じであるので、図 13 (d), (e) の上段に示すように、隣り合う三

角波の間に隙間が生じない。

【0104】なお、変数Length(n)の値が変数Width(n)の値を越えると、変数Width(n)で示される時間幅の中に三角波が収まらなくなるために出力波形を再生したときの音が不自然になるので、ステップS1419およびステップS1420において、変数Length(n)の値の上限が変数Width(n)の値に制限されている。

【0105】その後、ステップS1421に進み、関数Window1(n)および関数Window2(n)の増加率を表す変数W\_rate(n)の値が変数Length(n)の値の逆数に設定され、これにより、図13(d), (e)の上段に示すように関数Window1(n)および関数Window2(n)は、変数Length(n)の値だけ進む間に「1.0」高くなる鋸歯状波を表すこととなる。なお、上記で説明したように、子音部に対応する部分と母音部に対応する部分とは、変数Length(n)の値が相違するため、図13(d), (e)の上段に示すように、関数Window1(n)および関数Window2(n)の傾きは、子音部に対応する部分と母音部に対応する部分とでは相違する。

【0106】また、ステップS1421で、フラグレジスタF(n)の符号が逆転されて、フラグレジスタF(n)が示すチャンネルが換わり、その後、ステップS1422～ステップS1424において、チャンネルを示すフラグレジスタF(n)の値が正であるか負であるかに従って、カウンタPh1(n)および関数Window1(n)それぞれの値が設定されるか、あるいはカウンタPh2(n)および関数Window2(n)それぞれの値が設定される。これによって、図13(d)の上段および図13(e)の上段それぞれに示す波形の位相が互いに逆位相となるとともに、いずれの波形の周期も、関数Phase(n)が示す鋸歯状波の周期の約2倍となる。

【0107】また、ステップS1422～ステップS1424において、チャンネルを示すフラグレジスタF(n)の値が正であるか負であるかに従って、変数Start1(n)の値が変数Start2(n)の値に設定されるか、あるいは変数Start2(n)の値が変数Start1(n)の値に設定される。これによって、図13(d), (e)の上段に示す三角波と、波形データの区分との間の対応関係は、図13(c)に示す鋸歯状波がピークに達する度に、その時点における再生基準点の位置に従って更新されることとなる。例えば、図13(c)に示す鋸歯状波の各ピークうちの、右から3番目のピークの時点を考えると、図13には圧縮伸長率が「1.0」の場合が示されているので、再生基準点は、鋸歯状波の右から3番目のピーク位置から波形データの横軸へ垂線を延ばした位置にあり、区分Pitch3

に含まれている。従って、鋸歯状波の右から3番目のピークの時点から生じる、図13(d)に示す三角波対して区分Pitch3および区分Pitch4が対応付けられることとなる。このようにして、図13(d), (e)に示す、三角波と、波形データの区分との間の対応関係が得られる。

【0108】その後ステップS1425に進み、図13(f), (g)に示す出力データが形成される、以下説明する波形読出処理が行われ、DSPのメインルーチンに戻る。なお、前記ステップS1408において、変数Phase(n)は「1.0」で累増するようにして子音部での圧縮または伸長を行なわないようになっていたが、変数Tcomp(n)の関数で変数Phase(n)を累増するようにすれば子音部と母音部とで圧縮もしくは伸長の率を異ならせることができる。

【0109】図21は、波形読出処理ルーチンの前半部分のフローチャートである。図22は、波形読出処理ルーチンの後半部分のフローチャートであり、図21に示すステップS1611から図22に示すステップS1612へと処理の流れが続く。この波形読出処理ルーチンのステップS1601～ステップS1605では、図13(d)の上段に示す三角波を表す関数Env1(n)の値が求められる。

【0110】まずステップS1601において関数Window1(n)が変数W\_rate(n)の値だけ累増し、次にステップS1602～ステップS1605において、図13(d)の上段に示すように鋸歯状波を表す関数Window1(n)が高さ1で折り返された、三角波列を表す関数Env1(n)の値が求められる。即ち、関数Window1(n)の値が1よりも小さい範囲では関数Env1(n)の値は関数Window1(n)の値と同じ値となり(ステップS1603)、関数Window1(n)の値が1以上の範囲では関数Env1(n)の値は、「2-Window1(n)」となる(ステップS1604)。但し、関数Env1(n)の値がマイナスの値になるのは、以下の処理の都合上望ましくないので、関数Window1(n)の値が2以上の範囲では関数Env1(n)の値は「0」となる(ステップS1605)。

【0111】その後、ステップS1606～ステップS1608において、原波形の波形データを波形メモリの波形データ記憶領域(図2参照)から第1チャンネル用に読み込むためのアドレスがフラグレジスタVflag(n)の値に応じて計算される。即ち、ステップS1606においてフラグレジスタVflag(n)の値が、母音部に基づいて出力波形が形成されることを示す値「0」である場合には、変数Adress1の値は、変数Start1(n)の値に、カウンタPh1(n)の値と、フォルマント比率を表す変数Formant(n)の値との積が加えられた値となり(ステップS160

7)、フラグレジスタVflag(n)の値が、子音部に基づいて出力波形が形成されることを示す値「1」である場合には、変数Adress1の値は、変数Start1(n)の値に、カウンタPh1(n)の値が加えられた値となる(ステップS1608)。これにより、図13(d), (e)に示すように、三角波の底辺の長さに相当する時間が経過する間に、波形データの、その三角波に対応付けられた部分が順次読み込まれることとなる。

【0112】その後、ステップS1609に進み、変数Adress1の値が波形の末尾のアドレスを表す変数Endaddr(n)の値を越えた場合にはステップS1610において変数Gate(n)が「0, 0」に設定され、これによりボイスが消音される。その後、ステップS1611に進み、波形メモリの波形データ記憶領域の、変数Adress1が示すアドレスから波形データが読み込まれる。

【0113】その後、ステップS1612～ステップS1622において上記同様に第2のチャンネル用の各処理が行われる。説明は重複するので省略する。上記で説明したように、図13(d)の上段および図13(e)の上段それぞれに示す三角波は互いに逆位相であり、子音部に対応する部分では隣り合う三角波の間に隙間が生じないので、子音部に対応する部分では、関数Env1(n)の値と関数Env2(n)の値との和は常に「1. 0」となる。

【0114】ステップS1612～ステップS1622が終了した後、ステップS1623に進み、出力波形の波形データを表す変数Out(n)の値が次式によって求められる。

$$\text{Out}(n) = (\text{data1} * \text{Env1}(n) + \text{data2} * \text{Env2}(n)) * \text{Level}(n)$$

ここで、data1は、第1チャンネル用に読み込まれた波形データであり、data2は、第2チャンネル用に読み込まれた波形データである。

【0115】上式のうちのdata1 \* Env1(n)で表される部分が図13(f)に示されており、data2 \* Env2(n)で表される部分が図13(g)に示されている。なお、子音部に対応する部分では、上述したように、関数Env1(n)の値と関数Env2(n)の値との和は常に「1. 0」であり、また、子音部に対応する部分では、図13(d), (e)の下段に示すように、data1とdata2とは常に同一のデータである。従って、子音部に基づいて出力波形が形成される場合には、上式のうちの、data1 \* Env1(n) + data2 \* Env2(n)で表される部分は原波形の波形データそのものとなる。

【0116】また、ステップS1623では、変数Gate(n)の値に順次近づく関数L\_env(n)の値が計算される。即ち、変数Gate(n)の値と関数L

\_env(n)の現在の値との差に、関数L\_env(n)の値が変数Gate(n)の値に近づく程度を表す定数Kが掛けられた値に、関数L\_env(n)の現在の値が加えられた値が、関数L\_env(n)の新たな値となる。

【0117】その後ステップS1624において、ステップS1623で計算された変数Out(n)の値と関数L\_env(n)の値との積が、変数Out(n)の新たな値として代入されて、出力波形の波形データが得られ、図20に示す読出処理に戻る。以上説明したように、本実施形態の波形生成装置によれば、非周期的な振る舞いをする子音部を含む原波形に基づいて出力波形が形成される場合であっても、出力波形の、子音部に対応する部分は原波形そのままの波形となるので、自然な音色に聞こえる出力波形が形成される。

【0118】以下では図23～図28のフローチャートを参照しながら、本発明の波形生成装置の、上記で説明した動作と異なる他の動作について説明する。なお、図23～図28のフローチャートに示されている動作は全動作のうちの一部分の動作であるが、図23～図28のフローチャートに示されていない動作については全て上記で説明した動作と同じであるのでフローチャートを省略した。また、図23～図28のフローチャートに示されている動作であっても上記で説明した動作と同じ部分については重複説明を省略する。

【0119】以下説明する動作では、子音部において、時間軸方向の圧縮・伸長が行われず、母音部のみにおいて時間軸方向の圧縮・伸長が行われる。図23は、第2のボイスから第1のボイスへのリンク発音処理ルーチンのフローチャートである。

【0120】図24は、第1のボイスから第2のボイスへのリンク発音処理ルーチンのフローチャートである。図25は、第1のボイスの発音開始処理ルーチンのフローチャートである。図26は、第2のボイスの発音処理ルーチンのフローチャートである。図23～図26に示すフローチャートそれぞれは、図15～図18に示すフローチャートそれぞれに対応しており、図23～図26に示すフローチャートそれぞれは、1つ前のサンプリング周期におけるフラグレジスタVflag(n)の値を記憶するフラグレジスタVflag\_old(n)の設定が付け加えられている点を除き、図15～図18に示すフローチャートそれぞれと同様のフローチャートである。

【0121】図23および図24に示すフローチャートでは、一方のボイスから他方のボイスへフラグレジスタVflag\_old(n)の値が引き渡され、図25および図26に示すフローチャートではフラグレジスタVflag\_old(n)の値は「0」に初期設定される。図27は、読出処理ルーチンの前半部分を示すフローチャートである。

【0122】図28は、読み出し処理ルーチンの後半部分を示すフローチャートであり、図27に示す丸囲み記号の「4」から図28に示す丸囲み記号の「4」へと処理の流れが続く。図27および図28に示すフローチャートは、図19および図20に示すフローチャートに対応しており、図27および図28に示すフローチャートのうちの、図19および図20のフローチャートに示されている動作と同じ動作を行う部分であって説明を省略する部分については符号を省略した。

【0123】ステップS1416およびステップS1426において、フラグレジスタV<sub>f lag</sub>(n)の値が「1」であって、かつ、フラグレジスタV<sub>f lag</sub>\_old(n)の値が「0」であると判定され、即ち、再生基準点が初めて子音部に入ったと判定されると、図20に示すフローチャート同様にステップS1417に進む。以下説明は省略する。

【0124】ステップS1416およびステップS1426において、フラグレジスタV<sub>f lag</sub>(n)の値が「1」であって、かつ、フラグレジスタV<sub>f lag</sub>\_old(n)の値が「1」であると判定され、即ち、再生基準点が子音部内に存在し、初めて子音部に入った時点以外の時点であると判定されるとステップS1427に進む。

【0125】ステップS1427～ステップS1430では、関数Window1(n)および関数Window2(n)の増加率を示す変数W\_rate(n)の値が「0.0」に設定され、関数Window1(n)および関数Window2(n)の値が1または2に設定される。これによって、子音部に基づいて出力波形が形成される場合には、関数Env1および関数Env2は三角波列を表す関数とはならず、値「1.0」または値「0.0」の定数となる。また、子音部に基づいて出力波形が形成されている間はチャンネルの切換えは行われない。この結果、子音部に基づいて出力波形が形成される場合には、読み込まれた原波形の波形データが、そのまま出力波形の波形データとなる。

【0126】その後、ステップS1431に進み、フラグレジスタV<sub>f lag</sub>\_old(n)がフラグレジスタV<sub>f lag</sub>(n)の現在の値に更新される。これによって、ステップS1426において判定に用いられるフラグレジスタV<sub>f lag</sub>\_old(n)の値は、1つ前のサンプリング周期におけるフラグレジスタV<sub>f lag</sub>(n)の値を示すこととなる。

【0127】その後、ステップS1425に進んで波形読み出し処理が行われた後DSPのメインルーチンへ戻る。以上で説明した実施形態の波形生成装置によれば、非周期的な振る舞いをする子音部を含む原波形に基づいて出力波形が形成される場合であっても、出力波形の、子音部に対応する部分は原波形そのままの波形となるので、自然な音色に聞こえる出力波形が形成される。

### 【0128】

【発明の効果】以上説明したように、本発明の波形生成装置によれば、雑音のような非周期的な振る舞いをする部分を含む、言葉等を表す波形に基づいて、その波形とは音高等が異なる出力波形が再生される場合であっても、発音されたときの音が自然に聞こえる出力波形が形成される。

#### 【図面の簡単な説明】

【図1】本発明の一実施形態を示すブロック図である。

【図2】波形メモリ20の波形データ記憶領域を表す模式図である。

【図3】波形メモリ20の第1パラメータ記憶領域を表す模式図である。

【図4】波形メモリ20の第2パラメータ記憶領域を表す模式図である。

【図5】波形メモリ20の子音アドレス記憶領域を表す模式図である。

【図6】図1に示す操作子群60を示す図である。

【図7】CPU部用のプログラムのメインルーチンのフローチャートである。

【図8】再生処理ルーチンのフローチャートである。

【図9】図8のステップS403に示すルーチンの前半部分のフローチャートである。

【図10】図8のステップS403に示すルーチンの後半部分のフローチャートである。

【図11】ボイスモジュールへの割当処理ルーチンのフローチャートである。

【図12】DSPへの転送処理ルーチンのフローチャートである。

【図13】本発明の実施形態における波形データおよび関数の時間的变化を示す波形図である。

【図14】DSP部用のプログラムのメインルーチンのフローチャートである。

【図15】第2のボイスから第1のボイスへのリンク発音処理ルーチンのフローチャートである。

【図16】第1のボイスから第2のボイスへのリンク発音処理ルーチンのフローチャートである。

【図17】第1のボイスの発音開始処理ルーチンのフローチャートである。

【図18】第2のボイスの発音処理ルーチンのフローチャートである。

【図19】読み出し処理ルーチンの前半部分を示すフローチャートである。

【図20】読み出し処理ルーチンの後半部分を示すフローチャートである。

【図21】波形読み出し処理ルーチンの前半部分のフローチャートである。

【図22】波形読み出し処理ルーチンの後半部分のフローチャートである。

【図23】第2のボイスから第1のボイスへのリンク発

音処理ルーチンのフローチャートである。

【図24】第1のボイスから第2のボイスへのリンク発音処理ルーチンのフローチャートである。

【図25】第1のボイスの発音開始処理ルーチンのフローチャートである。

【図26】第2のボイスの発音処理ルーチンのフローチャートである。

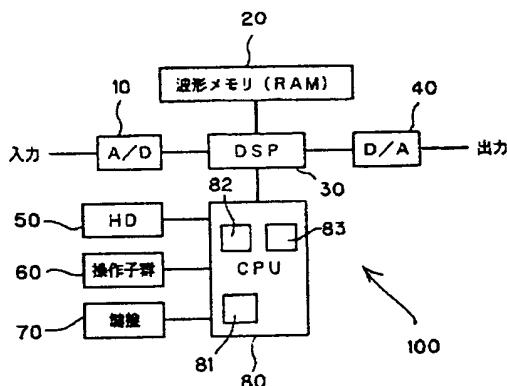
【図27】読み出し処理ルーチンの前半部分を示すフローチャートである。

【図28】読み出し処理ルーチンの後半部分を示すフローチャートである。

#### 【符号の説明】

10	A/D変換手段
20	波形メモリ
30	DSP部
40	D/A変換手段
50	ハードディスク
60	操作子群
70	鍵盤
80	CPU部
100	波形生成装置

【図1】



【図2】

<波形データ記憶領域>	
アドレス	データ
00000	wave data0
00001	
⋮	
07FFF	
08000	wave data1
08001	
⋮	
0FFF	
10000	wave data2
10001	
⋮	
17FFF	

<第1パラメータ記憶領域>	
アドレス	データ
0000	endadr0
0001	sadr0
0002	spitch0
0003	sadr1
0004	spitch1
⋮	
⋮	
endadr0	
0800	endadr0
0801	sadr0
0802	spitch0
0803	sadr1
0804	spitch1
⋮	
⋮	
endadr0	
1000	endadr0
1001	sadr0
1002	spitch0
1003	sadr1
1004	spitch1
⋮	
⋮	
endadr0	

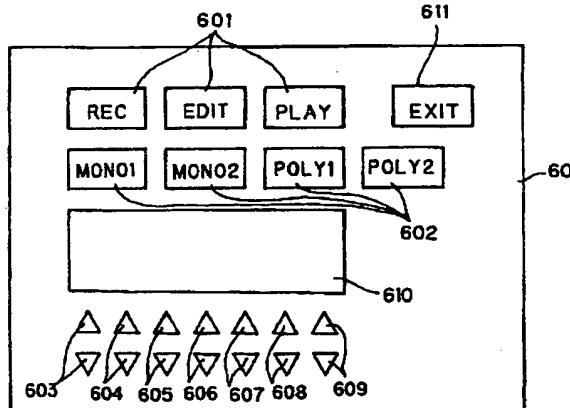
【図4】

【図5】

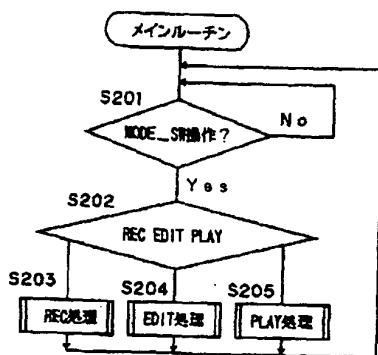
<第2パラメータ記憶領域>	
アドレス	データ
0000	org.pitch0
0001	org.pitch1
0002	org.pitch2
0003	org.pitch3
⋮	

<子音アドレス記憶領域>	
アドレス	データ
00000	vstart1
00001	vend1
00002	vstart2
00003	vend2
⋮	
0007F	
00080	vstart1
00081	vend1
00082	vstart2
00083	vend2
⋮	
000FF	
00100	vstart1
00101	vend1
00102	vstart2
00103	vend2
⋮	
0017F	

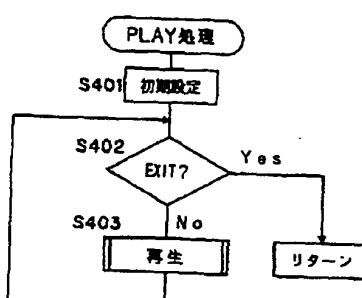
【図6】



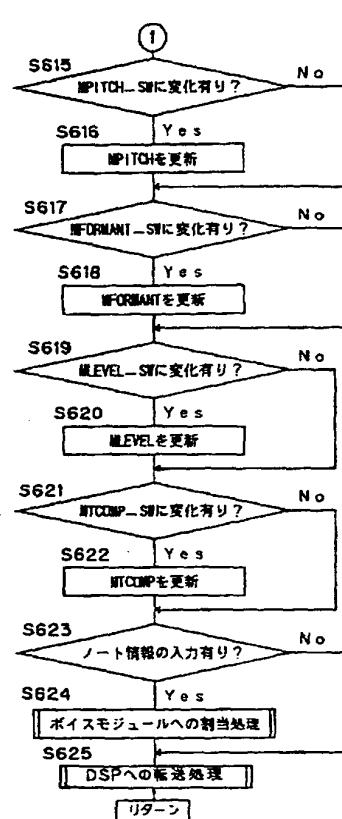
【図7】



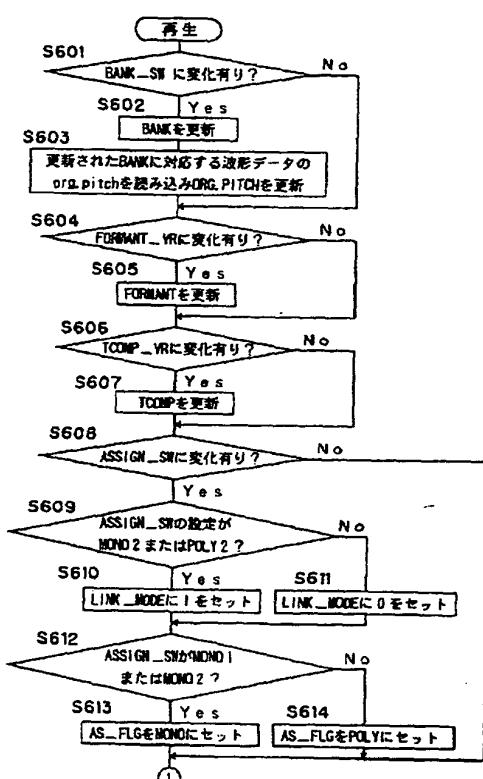
【図8】



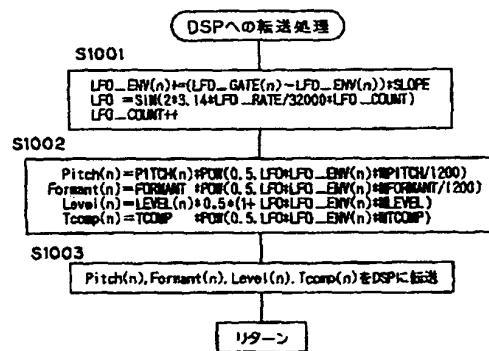
【図10】



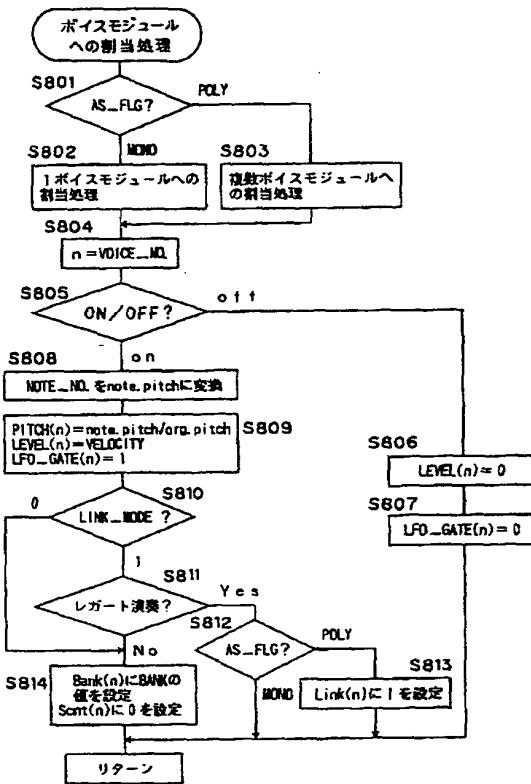
【図9】



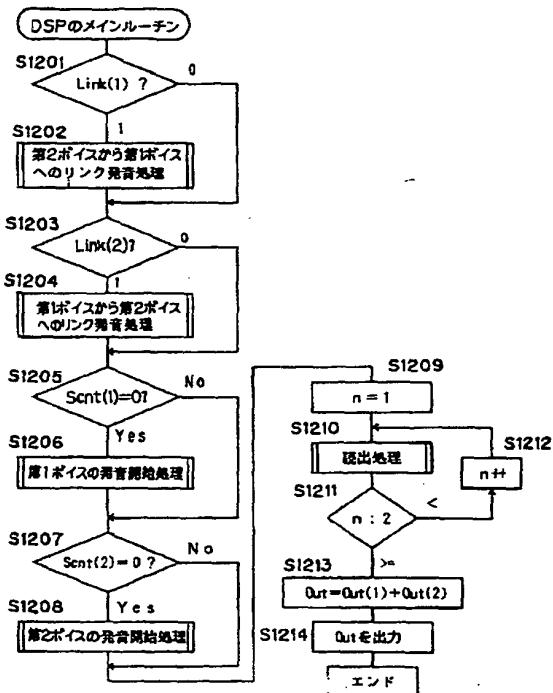
【図12】



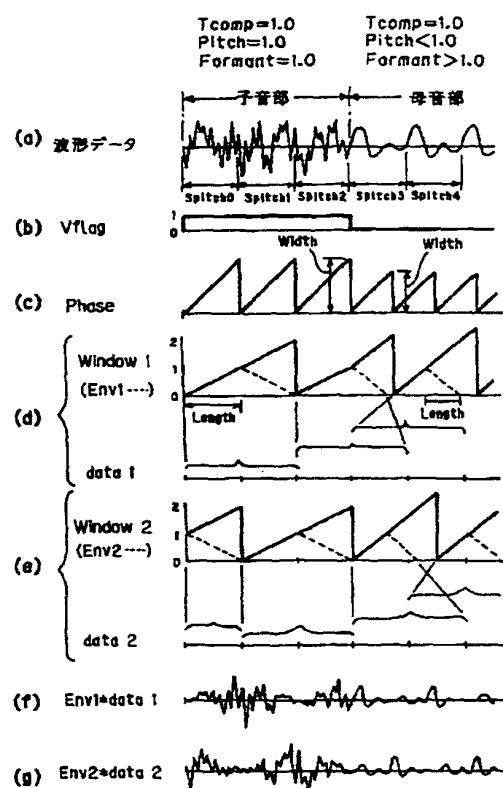
【図11】



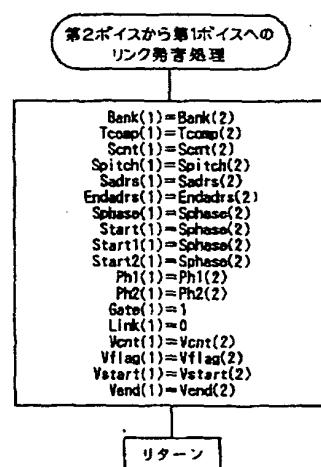
【図14】



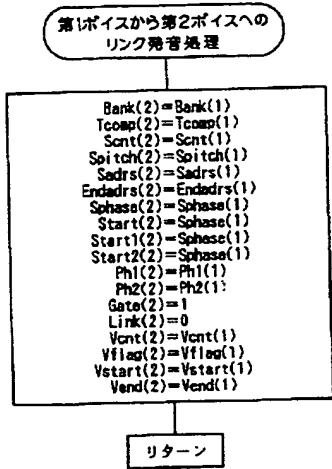
【図13】



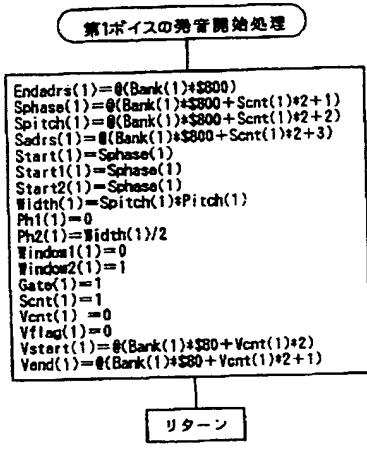
【図15】



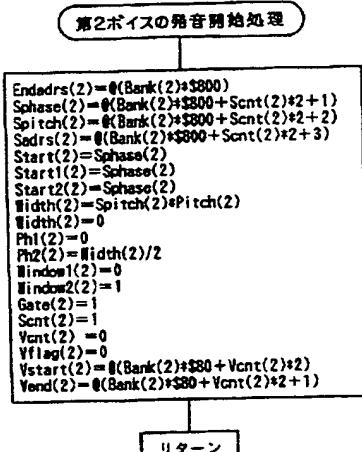
【図16】



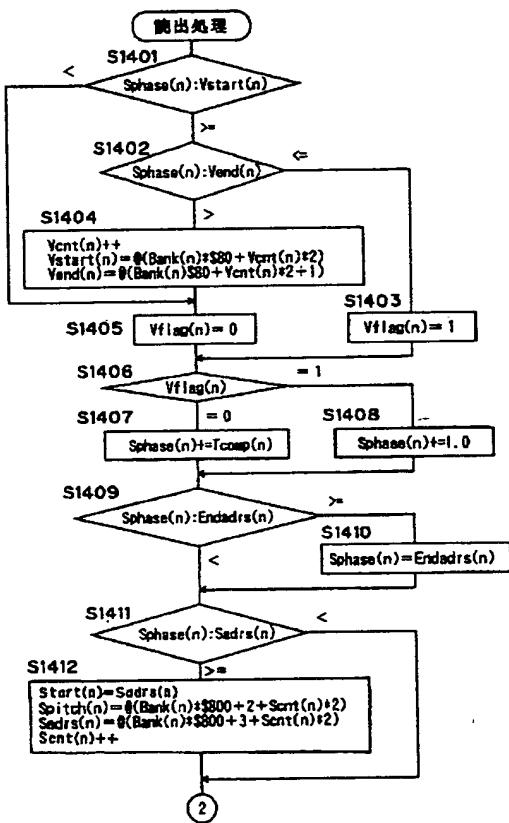
【図17】



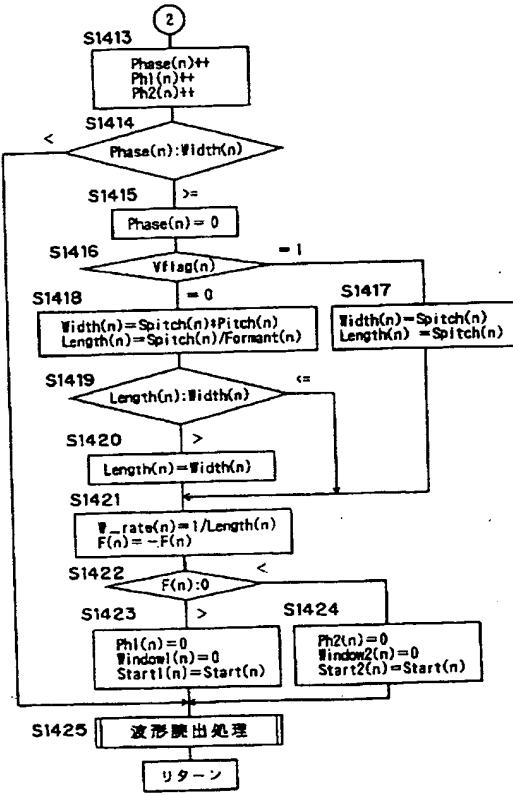
【図18】



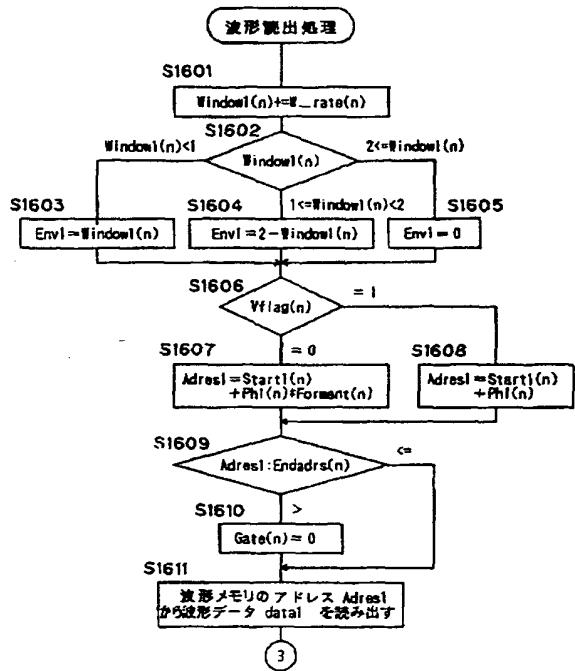
【図19】



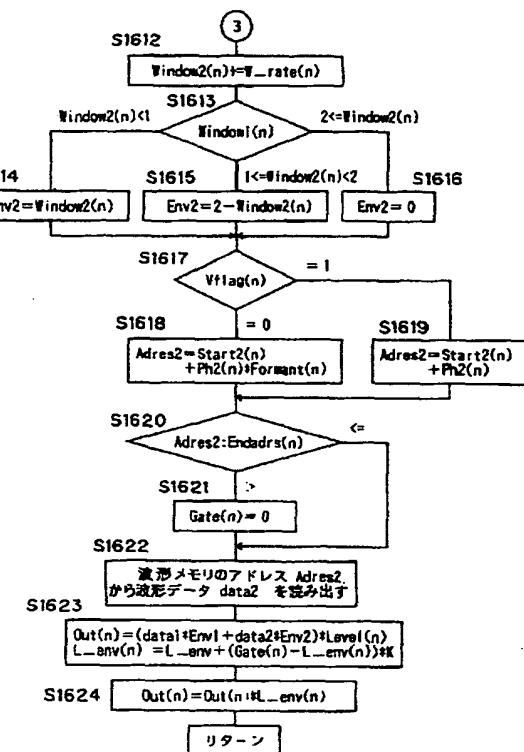
【図20】



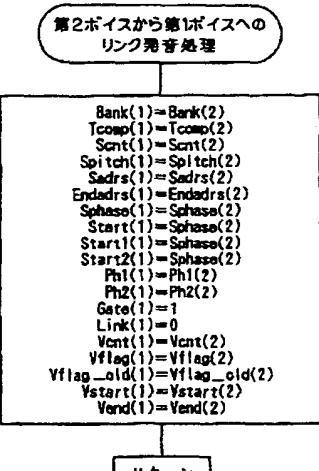
【図21】



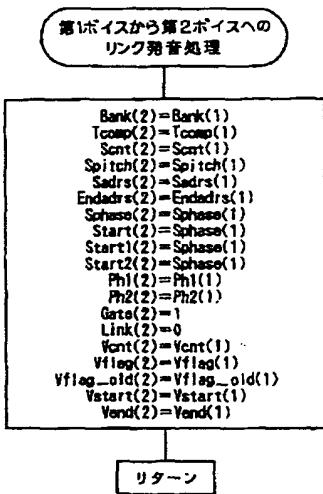
【図22】



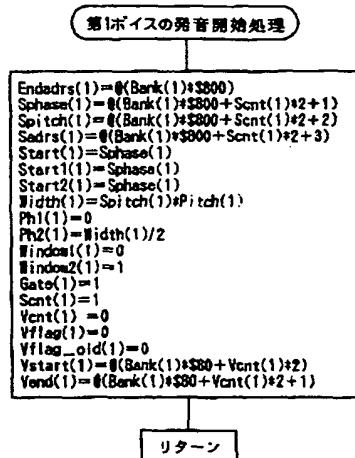
【図23】



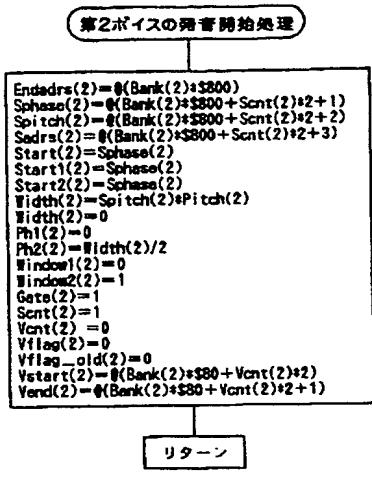
【図24】



【図25】

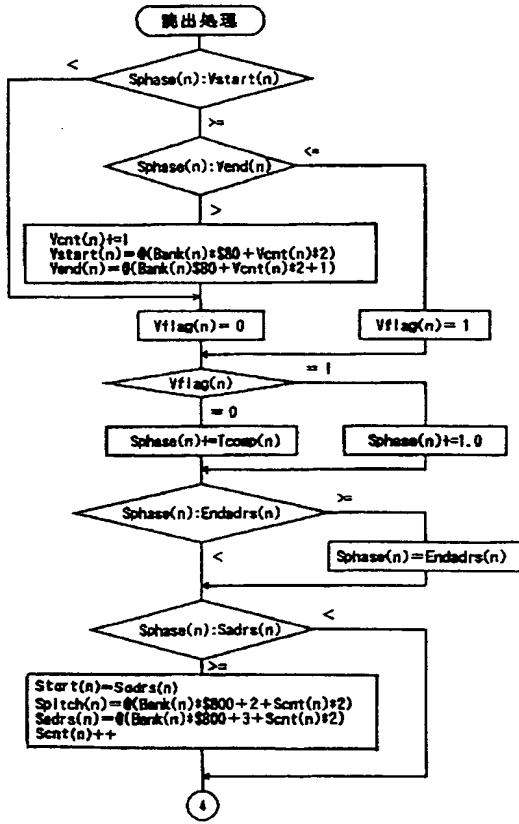


【図 26】



リターン

【図 27】



【図28】

